Tagging Wearable Accelerometers in Camera Frames through Information Translation between Vision Sensors and Accelerometers

Ali Akbari[1] Peiming Liu[2] Bobak J. Mortazavi[2] Roozbeh Jafari[1,2,3]
{aliakbari, peiming, bobakm, rjafari}@tamu.edu
[1]Department of Biomedical Engineering, Texas A&M University, College Station, Texas, USA
[2]Department of Computer Science and Engineering, Texas A&M University, College Station, Texas, USA
[3]Department of Electrical and Computer Engineering, Texas A&M University, College Station, Texas, USA

ABSTRACT

This paper presents a methodology to detect an object with an accelerometer potentially among many other moving objects in a camera scene. By matching sensor readings from a wearable accelerometer with analogous readings from a single camera or plurality of cameras, we detect instances of the same physical movement that both modalities capture. This has a wide range of potential applications in the cyber-physical systems domain such as identification, localization, and detecting context for activity recognition. We present an approach to project data from camera frames into accelerometer frames, where they share the same physical representation, allowing for comparing and determining similarities between the two modalities by using computational algorithms in the cyber world. This is challenging as depth is unknown when using a single 2D camera. We translate camera measurements into the acceleration physical domain and acquire an estimated depth, when the depth is not varying significantly during the motion. We model this translation as an optimization problem to find the optimal depth that maximizes the similarity between readings of the camera and accelerometer. Additionally, we discuss a potential solution with multiple cameras that works for arbitrary varying depth motions. Experimental results demonstrate that the system can detect matching between data stemming from physical movements observed by a wearable accelerometer and a single camera or plurality of cameras.

CCS CONCEPTS

Computing methodologies~Matching • Computing methodologies~Visual content-based indexing and retrieval

KEYWORDS

Signal Fusion, Camera, Wearable Sensors

ICCPS '19, April 16–18, 2019, Montreal, QC, Canada

ACM ISBN 978-1-4503-6285-6/19/04…$15.00
https://doi.org/10.1145/3302509.3311057

ACM Reference format:

Ali Akbari, Peiming Liu, Bobak J. Mortazavi, Roozbeh Jafari. 2019. Tagging Wearable Accelerometers in Camera Frames through Information Translation between Vision Sensors and Accelerometers. In *Proceedings of ACM International Conference on Cyber-Physical Systems ACM, Montreal, Canada, 11 pages.* https://doi.org/10.1145/3302509.3311057

1 INTRODUCTION

Tagging wearable accelerometers in camera scenes by detecting whether they measure the same movement can unlock many new privacy-aware sensing and computing paradigms. Cameras and wearable sensors, such as accelerometers, have received much attention in recent years due to their ubiquity and ability to support and enable a large number of applications. In particular, inertial measurement units (IMU) containing accelerometers and gyroscopes are becoming increasingly accessible in a variety of smart devices including smart watches [1]. Both accelerometers and cameras are widely used in different applications such as activity recognition, localization, and object tracking [2-4].

However, combined use of cameras and accelerometers has been limited because of challenges found in translation of signals from one sensing modality to another for fusing them together. Since the camera and accelerometer measure different aspects of human motion in different physical domains, translation of one modality to the other has continually inhibited development of new applications.

One such application enabled through hybrid use of camera and wearable accelerometer data would be privacy aware identification, wherein a user can enable the tracking of his/her wearable accelerometer by a camera merely by providing the sensor stream from the accelerometer to the cloud. Signal fusion techniques would leverage information captured from the cameras and the wearable accelerometer to identify which camera is observing the user and determine the boundaries of the movements of the wearable accelerometer in that camera's frame. This method of tracking would be privacy aware as the cameras would not need to process RGB information to perform facial or object recognition, but instead leverage the information collected from the accelerometer to identify and track the specific user. Another example is to monitor when a particular device with an integrated accelerometer leaves the environment without authorization, which is an important asset tracking application.

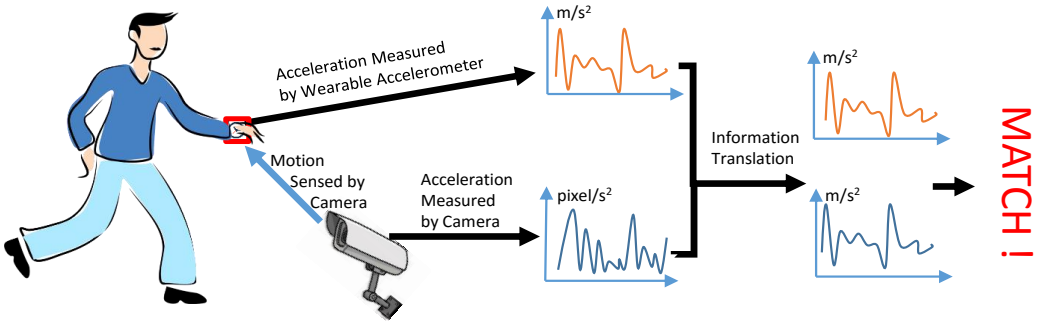

Figure 1. The overview of the matching algorithm. Both the camera and the wearable accelerometer capture the same physical movement, but it is expressed with different units and patterns. Information translation successfully brings these two modalities into the same space where they share same units and similar patterns

In these applications, the challenge is to find the match between cameras and an accelerometer and then tag the accelerometer in the view of those cameras. Such a technique can also enable cameras to only focus on important parts of the video frames for image processing by ignoring the irrelevant segment or segments that may carry sensitive information and are subject to privacy concerns. These are merely a few of many potential examples. Effective matching of cameras and wearable accelerometers may likely enable many more such applications.

The fact that cameras and wearable accelerometers offer completely dissimilar observations of the same physical movement constitutes the major difficulty in identifying their relationship. Cameras capture movements in a two-dimensional space, with kinematics represented as changes in pixels and with the frame of view and reference remaining potentially unchanged. Accelerometers, on the other hand, capture movements in a 3-dimensional space, with kinematics represented in m/s^2. Additionally, in the case of accelerometers, both acceleration due to gravity and due to a subject's motion are captured [2]. The frame of view or reference of accelerometers also changes due to the placement of wearable sensors on moving limbs. These challenges impede effective and direct comparison of data captured from cameras and accelerometers when trying to determine whether both modalities are reporting the same movement. Thus, algorithms are required to intertwine the physical models of the sensor and the camera in order to bring the two sensor modalities into the same physical or cyber domain, so that they can be compared to each other.

Most prior works investigating fusion of cameras and wearable accelerometers simplify these challenges by requiring either a fixed or known placement of the sensors or fixed or known movements [4]. The fixed or known placement of wearable accelerometer and cameras assists in determining the relative position between the frames of the two modalities to translate one modality to another. This places burden on users by asking them to follow certain placement instructions or to provide the placement information to the system. Another approach is to limit the users to only particular movements, such as walking and waving a hand. With these restrictions, the techniques build and use movement models leveraging unique and gesture-specific features in order to recognize matching between the modalities. These limitations largely restrict the potential of this technology and fail to establish a strong relationship between two modalities.

In this paper, we focus on detecting whether the data captured by a camera or plurality of cameras and an accelerometer describe the same movement, but not focus on object tracking algorithms using cameras. There are many object tracking algorithms based on different notions, such as optical flows [5-6]. We assume that the algorithm used to track objects in camera frames works well, the camera is stationary, and its placement is known, but there is no need to fix the location or orientation of the accelerometer.

The overview of our proposed method is illustrated in Figure 1. Our approach brings these two physical modalities into the same space, which is the domain of physical acceleration, and enables direct comparison between the camera and accelerometer data. With multiple RGB cameras, it is possible to recover a 3-dimensional model of an object and from that calculate the 3D acceleration in order to compare it with the accelerometer's direct measurement. The depth information is also available in case of using depth cameras such as Microsoft Kinect. However, with a single RGB camera, it becomes a very difficult task since it is not possible to retrieve depth information. To address this issue we propose a method to use a single camera to estimate depth for motions in which depth variation is limited. We do not need to define and use higher-level features specific to certain kinds of movement, instead we rely only on the raw data. Moreover, a possible solution in the multiple camera scenario is given that works for arbitrary motions that may include large depth changes. Using acceleration as the fingerprint of a movement, this method finds if the camera is seeing the movement measured by the accelerometer through the information translation.

The contributions of this paper are the following:

- An algorithm is proposed to detect whether a single camera and an accelerometer are observing the same motion while depth is unknown. The solution is offered for two cases including constant depth and limited varying depth. The method does not require knowledge of orientation or a specific placement of accelerometers.
- For the case of constant or limited varying depth, even using a single camera, this method can also estimate the depth of the object with respect to the camera.
- The potential solution for the general case of using multiple

cameras is discussed, which can work for any arbitrary movement. This method takes advantage of gravity and uses it as a key to bring cameras and accelerometers into the same space, thus avoiding the use of complicated filters to eliminate gravity from acceleration.

- Performance of the proposed method is tested experimentally with an inexpensive Sony PS Eye camera and wearable accelerometers.

The remainder of this paper is organized as follows. The related works are reviewed in Section 2. The preliminaries and the proposed algorithm are introduced in Sections 3 and 4 respectively. Experimental results are illustrated in Section 5 followed by a conclusion in Section 6.

2 Related Works

Many researchers fused cameras and accelerometers with the goal of accurate position or orientation estimation as well as localization [2,7-10]. Although this type of data fusion needs to bring the two modalities into the same domain in order to to combine their measurements, its ultimate goal is different from our work which is tagging the accelerometer in the camera frames. Multiple solutions based on the fusion of these two modalities were proposed to solve the simultaneous localization and mapping (SLAM) problem in robotics [11-12]. A typical camera and accelerometer based solution to the SLAM problem estimates the pose and location of a robot. Researchers presented a localization, mapping and self-calibration algorithm for visual and inertial sensors including accelerometer and gyroscope [13]. They applied an unscented Kalman filter (UKF) and tried to estimate the real pose and the motion of sensors over time. There are very few methods that have performed the fusion of camera and inertial sensor measurements without a filter-based approach. For instance, a batch method that performs SLAM from image and inertial measurements addressed the data fusion problem by minimizing a cost function using the Leven-Marquardt algorithm [14]. A closed-form solution for orientation, speed, and accelerometer bias determination by fusing camera and accelerometer data was proposed based on a non-standard observability analysis to considers system nonlinearities [15].

Data fusion of camera and accelerometer has provided opportunities in augmented reality (AR) and virtual reality (VR) fields as well [16-17]. In AR, the main challenge is to accurately estimate parameters such as velocity, position, and orientation of human subjects. In prior work, researchers integrated a camera and an inertial sensor including gyroscope and accelerometer into a single device, providing a hardware-synchronized stream of video and inertial data. The investigators used an extended Kalman filter (EKF) to fuse these two modalities to obtain camera pose tracking in real time [17]. The goal of these techniques is to enhance the accuracy of the algorithms by leveraging information from different modalities.

On the other hand, tagging accelerometer in the camera frame, which is the main goal of the current study, mainly answers the following question: is the data stream from a particular accelerometer matched with an object seen by the camera? The correlation between signals from the camera and accelerometer worn by the person was used to identify a person out of many people in a camera view [18]. This method requires some knowledge about the relationship between the coordinate systems of the camera and the accelerometer. A different investigation achieved object matching between the two modalities by using an inverted pendulum model of human gait to model walking [4]. By attaching a smartphone to a user's belt, the investigators were able to record acceleration and could further generate speed from the acceleration using the inverted pendulum model and matched it to the speed sensed in the camera frame. Although the proposed method offers high accuracy, it only handles walking, and the accelerometers have to be placed at a certain location with a certain orientation. Both [18] and [4] assume that the objects' depth is constant, and they do not discuss the challenges associated with varying depths. Researchers stressed the challenges brought by gravity interference from accelerometers, and attempted to compensate for it by adding the same effect to the cameras [19]. The correct value of gravity in camera data can only be computed when the distance of the object to the camera (also known as depth) is known. However, with a single regular RGB camera the depth is unknown. We overcome this limitation by estimating the depth automatically for motions that experience constant or limited-varying depth changes by modeling it as an optimization problem in which the similarity between readings from the camera and accelerometer is maximized.

3 Preliminaries

In this section, we present the preliminaries required to identify a match between a wearable accelerometer and cameras. In other words, the algorithm should detect whether or not the motion measured by a specific accelerometer is seen by a camera. Our framework consists of two major components: 1) A tracking algorithm, which tracks objects in the cameras' frames and calculates their accelerations, and 2) A tagging algorithm that detects if the motion of the tracked objects is matched with the motion sensed by an accelerometer. The proposed algorithm is a general-purpose matching algorithm, and requires minimal prior information about the setup and deployment details. Additionally, the proposed method can work with accelerometers typically available in smartphones, which makes it ubiquitously available. It also exhibits fast convergence time, thus enabling real-time operation. We use the Lucas-Kanade (LK) method [6] as our optical tracking algorithm and build our matching algorithm based on that. However, theoretically, any tracking algorithm should work as long as it exhibits similar or higher accuracy than the LK algorithm.

3.1 Input Instances

Our method receives data from cameras and a wearable accelerometer. These two inputs are modeled as follows:

3.2.1 Input from Cameras. During the observation time from t_1 to t_n ($n > 1$), n frames are captured by the camera; the timestamp for frame j ($n \geq j \geq 1$) is t_j. The tracking algorithm generates sets of points $P_i = \{p_{i,j}\}$, where $p_{i,j}=(x, y)$ represents the coordinates of object i in frame j at time t_j. We then calculate the acceleration from these points. We denote the acceleration that is extracted

from camera information for object i at frame t_j as $a^c_{i,j} = [a^c_{x,i,j}, a^c_{y,i,j}]^T$, where a^c_x and a^c_y are accelerations in x and y directions in the 2D camera frame. Our approach attempts to translate cameras information into the accelerometer's space by adding the effect of gravity interference to data obtained from the cameras (Section 4), so we denote the gravity-induced acceleration in the camera as $\tilde{a}^c_{i,j}$ for object i at t_j.

3.2.2 Input from Cameras. During the same observation time from t_1 to t_n, more than n readings are generated by the accelerometer in our setup, because our cameras capture frames at 30Hz while accelerometers provide readings at 60Hz. We pick the closest reading to each timestamp t_j when the frame j is recorded to synchronize the two data streams. We pick the earlier data point if there are two readings equally close to t_j. We denote the synchronized acceleration sensed by accelerometer i as $a^m_{i,j}$, where $a^m_{i,j} = [a^m_{x,i,j}, a^m_{y,i,j}, a^m_{z,i,j}]^T$ represents the accelerometer-sensed acceleration of object i closest to time t_j in x, y and z directions of the local accelerometer frame.

3.2 Camera to accelerometer Translation Framework

The heart of our proposed method is translating camera readings into the same space as the accelerometer, where both modalities share the same units (m/s^2), physical meaning, and dimensions. Frames generated by cameras are first processed by the LK object tracking algorithm to generate the trajectory of moving objects. Using tracks calculated by the optical object tracking algorithm, the camera-sensed acceleration is calculated (Section 4.1). After synchronization between camera and accelerometer readings, we use the synchronized data as inputs to the proposed matching algorithm. For a single camera scenario, camera-to-sensor translation (CST) is determined and used as the key method to bring the two modalities into the same space, where they can be compared to each other (Section 4.3). This method can restore the depth information by solving an optimization problem for movements that experience constant or limited-varying depth to the camera. A multiple-camera solution may also be utilized to restore depth information in arbitrary depth-varying motions and offers direct comparison between the two modalities (Section 4.3). Figure 2 describes the overall workflow of our proposed approach.

4 Camera-accelerometer matching Algorithm

In this section, we describe the technical details on how to detect matching between readings from a camera and a wearable accelerometer by using information translation technique between the two modalities. We first introduce our approach using a single camera when objects are moving at a fixed depth and extend it to a limited depth- varying situation afterward. The general fusion algorithm that uses multiple cameras is discussed in Section 4.3. Based on the knowledge that accumulated error will become intolerable for accelerometers during the process of integration to obtain velocity or displacement, acceleration is directly used as the main feature to find the match between these two modalities.

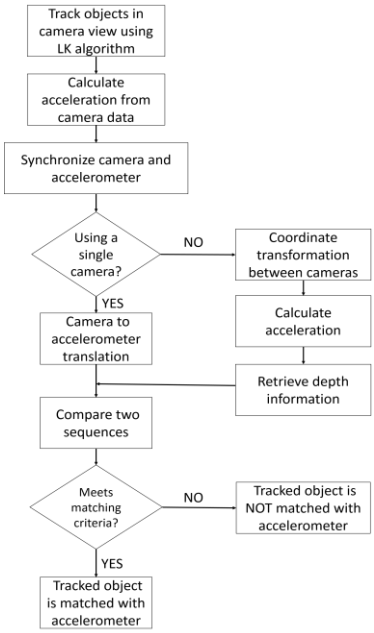

Figure 2. Overall workflow of the proposed method

4.1 Extracting Linear Acceleration from a Camera

We first need to estimate the acceleration $a^c_{i,j}$ of object i at each timestamp t_j from its tracked position $P_i = \{p_{i,j}\}$ observed by the camera; assume object i is moving with a uniformly accelerated motion from t_{j-1} to t_{j+1}. This assumption is valid as long as the frame rate of the camera is sufficiently high. The lower bound required for the frame rate is about 20Hz based on our experiments on normal human movements. Based on this assumption, $a^c_{i,j}$ for object i in the camera view at time t_j can be calculated using Equation 1.

$$a^c_{i,j} = \begin{bmatrix} a^c_{x,i,j} \\ a^c_{y,i,j} \end{bmatrix} = 2\frac{\Delta t_j \cdot \Delta p_{i,j+1} - \Delta t_{j+1} \cdot \Delta p_{i,j}}{\Delta t_j \cdot {\Delta t_{j+1}}^2 + \Delta t_{j+1} \cdot {\Delta t_j}^2} \quad (1)$$

where

$$\Delta t_k = t_k - t_{k-1}$$
$$\Delta p_{i,k} = p_{i,k} - p_{i,k-1}$$

4.2 The Relationship between Pixels and Distance in Meters

We use the pinhole camera model, which is a simple model for analyzing the camera's geometric properties, to establish the relationship between pixels and distance in meters [20]. In Figure 3, an object of size x meters at depth d meters from the camera's optical center is observed in y pixels in a camera frame out of the Y total pixels of the field of view. f represents the focal length in meters and α is the angle of view, both of which are known parameters of a camera. Using this model, the relationship between y and x can be established by Equation 2.

$$x = \frac{2\tan\frac{\alpha}{2} \cdot d}{Y} \cdot y \quad (2)$$

4.3 Single-Camera Matching Algorithm through Camera-To-Sensor Translation

Our proposed approach for a single camera tries to build the translation between the camera and wearable accelerometer so that they can be compared in the same physical space with the same unit and physical meaning. We establish the camera-to-sensor translation (CST), which is discussed later in this section, by finding the physical length of pixels in the image frame.

4.3.1 Establish Translation for an Object Moving at Constant Depth Relative to a Camera. Equation 2 reveals a linear relationship between an object's depth d and its actual size x under the observation of pixel length y in an image frame. However, d is an unknown variable. While one can recover a 3-dimensional model of an object using multiple cameras through complex vision algorithms, a single regular RGB camera alone without depth sensors is not capable of capturing depth information.

To address the above issue, we develop a model to recover depth information of objects automatically by fusing data that comes from both camera and wearable accelerometer. This solution assumes that the direction of motion is parallel to the image plane of the camera; in other words, d is assumed to remain constant here. Later, we expand this solution to motions with limited depth variation as well. However, if d varies a lot during a movement, it will be impossible to retrieve depth information with a single regular camera. In such case, a depth camera or multiple cameras will be required.

In our proposed solution, instead of trying to filter out gravity interference as a source of noise from the accelerometer, we leverage it as a bridge to bring the two modalities into the same physical domain. To ensure the camera and accelerometer measure the same physical movement with analogous physical meaning, the first step of translation is to add a similar induced gravity to the camera readings. For a camera set up as shown in Figure 4, with θ degrees between its x-axis and horizon, gravity in the camera's coordinate system is divided into two parts g_x and g_y. Using the relationship shown in Equation 2, Equation 3 can then be used to quantify gravity interference g_{cam} in the camera.

$$g_{cam} = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \|g_{cam}\| \cdot \sin\theta \\ \|g_{cam}\| \cdot \cos\theta \end{bmatrix} \tag{3}$$

where

$$\|g_{cam}\| = \frac{Y}{2\tan\frac{\alpha}{2}\cdot d} \cdot \|g_{phy}\|_{pixel/s^2}$$

$\|g_{phy}\|$ is the acceleration due to gravity on earth, which is 9.8 m/s^2. To ensure the analogous physical meaning of these two modalities, we add gravity interference g_{cam} to the camera-sensed acceleration $a^c_{i,j}$ at each timestamp t_j. Since accelerometer-sensed acceleration ($a^m_{i,j}$) already includes gravity, as shown in Equation 4, we similarly calculate gravity-induced camera-sensed acceleration $\tilde{a}^c_{i,t}$ as in Equation 5 by incorporating g_{cam}.

$$a^m_{i,j} = g_{phy} - a_{linear}{}_{m/s^2} \tag{4}$$

$$\tilde{a}^c_{i,j} = g_{cam} - a^c_{i,j}{}_{pixel/s^2} \tag{5}$$

Even though Equation 5 offers analogous observations from the camera and accelerometer by adding the same gravity interference, to generate gravity-induced camera-sensed acceleration $\tilde{a}^c_{i,j}$, the value of g_{cam} needs to be known, and the units of the camera-sensed and accelerometer-sensed accelerations need to be the same; both of the requirements can be met only when the depth d is known. To acquire an estimated depth d, we model it as an optimization problem to find the optimal depth d that maximizes the similarity between readings from the camera and accelerometer. Based on Equation 2, the ratio between the actual size of the object and the size of its projection into the camera frame is a constant value at a certain depth. Thus, as long as the depth is constant, the ratio between the gravity measured by the camera and accelerometer is same as the ratio between the motion acceleration measured by these two modalities, as Equation 6 shows.

$$\frac{\|g_{phy}\|}{\|g_{cam}\|} = \frac{\|a^m_{i,j}\|}{\|\tilde{a}^c_{i,j}\|}. \tag{6}$$

From Equation 6, we derive Equation 7, where ω scales camera-sensed motion acceleration from pixels to meters. We refer to ω as the outcome of CST, and it is a function of depth d.

$$\|\tilde{a}^c_{i,j}\| \cdot \omega(d) = \|a^m_{i,j}\| \tag{7}$$

where

$$\omega(d) = \frac{\|g_{phy}\|}{\|g_{cam}\|} = \frac{2\tan\frac{\alpha}{2}\cdot d}{Y}$$

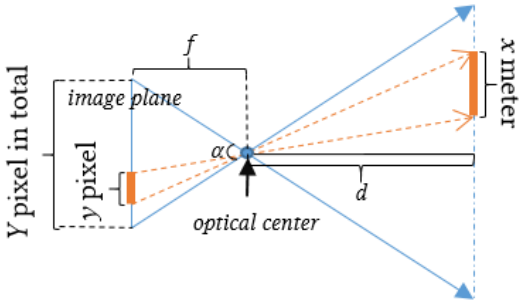

Figure 3. Pinhole Camera model

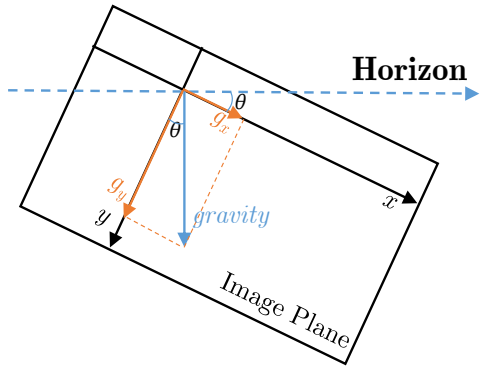

Figure 4. A camera placed with θ degree between its x-axis and the horizon in the physical world

We then model our estimation as an optimization problem as follows:

PROBLEM 1. Find the optimal **CST**. Let $a_{i,j}^m$ and $a_{i,j}^c$ be synchronized time sequences of accelerations from an accelerometer and a camera respectively, and $\tilde{a}_{i,j}^c$ be the gravity-induced camera-sensed acceleration. The optimal CST, ω, is determined by minimizing the distance between $\|\tilde{a}_{i,j}^c\| \cdot \omega(d)$ and $\|a_{i,j}^m\|$. Using the L_2 norm to measure the distance, an optimization problem is formulated as shown in Equation 8:

$$d = argmin\, h(d) = \underset{d}{argmin} \sum_{j=1}^{n} (\|\tilde{a}_{i,j}^c\| \cdot \omega(d) - \|a_{i,j}^m\|)^2 \quad (8)$$

where

$$\|\tilde{a}_{i,j}^c\| = \|a_{i,j}^m\| = (a_{x,i,j}^m{}^2 + a_{y,i,j}^m{}^2 + a_{z,i,j}^m{}^2)^{\frac{1}{2}}$$

$$\|g_{cam}\| = \frac{9.8}{\omega(d)}$$

By solving the optimization problem in Equation 8 for d, we can estimate the depth of the object with a single camera and the accelerometer data. However, since $\|g_{cam}\| = 9.8/\omega(d)$, we can replace $\omega(d)$ with $9.8 \,/\, \|g_{cam}\|$ and treat $\|g_{cam}\|$ as the only unknown. This allows us to avoid specifying parameter α to our algorithm if the depth (*i.e.*, the value of variable d) is not important for the target application. α is a constant parameter of the camera and is required for calculating d from $\omega(d)$ based on Equation 7. Equation 8 specifies a least-sum-of-squares optimization, which is solvable by the Gauss-Newton or gradient descent methods [21].

4.3.2 Matching Criteria. After finding the optimal CST (*i.e.*, ω or $\|g_{cam}\|$), which minimizes the function $h(d)$ in Equation 8, we calculate the distance between readings from the camera, after translations by applying $\omega(d)$, and the accelerometer. The resulting distance (*i.e.*, $\min h(d)$ is the translated distance between these two sequences and allows us to design a criteria to determine whether the two sequences are matched (*i.e.*, coming from the same motion).

Setting a constant threshold on translated distance is the simplest method since a smaller translated distance always implies a higher similarity between two sequences. However, finding a constant threshold that covers most cases and rarely leads to misdetection is challenging. In our experiments, the translated distance between two sequences is greatly affected by the type of movements; fast movements always lead to larger translated distances than slow movements because of larger noise. Therefore, fast movements may need a larger threshold compared to slow movements. As a result, it is challenging to determine a constant threshold for all possible movements. To address this problem, we design an adaptive threshold. We formulate our criteria as follows:

$$\begin{cases} \lambda < \rho & \rightarrow matched\ sequences \\ \lambda \geq \rho & \rightarrow irrelevant\ sequences \end{cases}$$

where

$$\lambda = \frac{min(h(\|g_{cam}\|))}{\mu} \quad (9)$$

$$\mu = \lim_{\|g_{cam}\| \to \infty} f(\|g_{cam}\|) = \sum_{t=t_1}^{t_2} (9.8 - \|a_{i,j}^m\|)^2 \quad (10)$$

where $\rho \leq 1$ is a constant value specified by the user and serves as a threshold, but λ is calculated based on the accelerometer data. The proposed criteria adjusts itself according to the acceleration of movements as it takes advantage of readings from the accelerometer to adjust the value of μ. Fast movements lead to larger values of μ that alleviates effect of large $h(\|g_{cam}\|)$. Details of driving this adaptive threshold is shown in Appendix 7.1.

4.3.3 Establish Translation for Objects Moving at Limited Varying Depth Relative to the Camera. Dealing with the depth changing scenarios by using a single camera, perspective projection distortion and loss of depth changing information are two major challenges [22]. The camera loses depth changing information during the projection process. Loss of depth changing information occurs when objects are projected from a three-dimensional space to a two-dimensional image. In this section, we assume that objects experience limited depth-varying movements. The limitation is that the direction of the movement should be preserved. Under this assumption, we can add gravity interference in a fixed direction as we did in the constant depth scenario. The camera only observes movements projected onto the image plane, while the accelerometer measures movements in all three directions. To compare only the parts of movements that are sensed by both modalities, we use Equation 11 as a filter to remove unnecessary information from the original accelerometer's readings $a_{i,j}^m$, and obtain filtered acceleration $\hat{a}_{i,j}^m$. In Equation 11, $v=[x,y,z]^T$ is a constant unit vector in the accelerometer's frame with the same direction as the acceleration not observed by the camera (camera cannot observe depth changing) and $\delta(v)$ is a value between 0 and 1 and acts as the filter.

$$\hat{a}_{i,j}^m = a_{i,j}^m \cdot \delta(v)$$

$$\delta(v) = \sqrt{1 - \left(\frac{(a_t^m)^T . v}{\|a_{t\,i,j}^m\|}\right)^2} \quad (11)$$

Using the filtered acceleration, we extend Equation 8 to Equation 12 to adopt depth-varying scenarios.

$$d, v = argmin\, h(d, v) = \underset{d,v}{argmin} \sum_{j=1}^{n} (\|\tilde{a}_{i,j}^c\| \cdot \omega(d) - \|a_{i,j}^m\| \cdot \delta(v))^2 \quad (12)$$

Equation 12 has a clear physical meaning where we measure the similarity between these two modalities in the same space after the optimal CST (*i.e.*, ω) translates pixels into meters and the filter δ removes the unobservable part of accelerations from accelerometers. To solve Equation 12 and to find the best ω and δ, we follow a similar process as discussed in the constant depth scenario. We reuse the decision criteria discussed in the previous section as well. To completely address the challenges associated with varying depth movements and in order to expand this work to arbitrary motions, we adopt a multiple camera scenario and discuss the corresponding methodology in the next subsection.

4.4 Multiple Cameras Matching through Depth Information Recovery.

With a single camera, depth information is naturally lost and is challenging to restore. Thus, in the previous section with a single camera, we first investigated the constant depth paradigm, which allows us to estimate the depth using a distance minimization approach, and then we limited the varying depth movements by considering motions in which the direction is not changed to avoid intolerable distortion. This limitation however can be released by utilizing multiple cameras. In this section, we discuss a general solution to the problem of detecting matching between accelerometer and camera measurements leveraging two cameras. In contrast to the single camera paradigm, with the multiple camera paradigm the depth can be determined, using Equation 1, acceleration can be directly calculated from the camera readings, and then it can be compared to the accelerometer readings.

In the two-camera scenario, each pixel in a camera frame corresponds to a camera ray. By using two or more cameras, the 3-D location of the object can be determined by leveraging the unique intersection point of multiple camera rays, and thus the depth is recovered [23]. We can thus use this information to restore the depth. We continue to use the pinhole camera model to analyze the properties of the cameras.

We select one of these cameras and use its frame as the global frame called G. A vector v observed in other camera's i^{th} frame F_i can be transformed into v' in G using Equation 13.

$$v' = Rv + T \tag{13}$$

$\boldsymbol{R}$ in Equation 13 is the rotation matrix, and $T = [x, y, z]^T$ is the translation matrix, which translates the origin $\boldsymbol{o}$ of the frame F_i to $o' = (x, y, z)$ in the global frame G. Using Euler angles ψ (roll), θ (pitch), and φ (roll) to denote its relative orientation towards global frame G, the final status of the frame is obtained. The value of rotation matrix $\boldsymbol{R}$ relative to G is shown in Appendix 7.2.

To model the camera ray corresponding to pixel $p=(x, y, f)$, where f is the focal length, we use Equation 14 to describe it in the global frame G. o' in Equation 14 is the origin of the ray in global frame G, and τ is the parameter of the function.

$$r(\tau) = o' + \tau(\boldsymbol{R}p + T) \tag{14}$$

Ideally, if we have two camera rays capturing the same object via different lenses, they should intersect, and the point of intersection is the physical location of the object in the global frame (Figure 5). However, noise from the object tracking algorithm may inhibit the detection of the point of intersection. To obtain a reasonable estimation, we find a pair of points on the rays that have the shortest distance among all pairs of points and use the midpoint of the line between these two points as an estimation of the intersection or location of the object. Thus, we can estimate the depth of the object. To find a pair of points between the camera ray $r_1(\tau_1) = o_1 + u\tau_1$ and camera ray $r_2(\tau_2) = o_2 + v\tau_2$, we denote vector $e = r_1(\tau_1) - r_2(\tau_2) = o_1 - o_2 + \tau_1 u - \tau_2 v$ as the vector from a pair of points on two rays as illustrated in Figure 5.

The length of the vector e should have its minimum value at a unique pair of points p_1 and p_2 if the camera rays r_1 and r_2 are not parallel. Specifically, the distance between p_1 and p_2 will reach its minimum value when vector e is simultaneously perpendicular to both r_1 and r_2, and this property will lead to Equation 15.

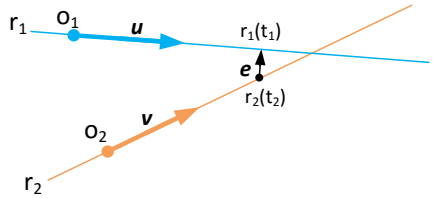

Figure 5. The pair of points that have shortest distance

$$\begin{aligned} u \cdot e &= 0 \\ v \cdot e &= 0 \end{aligned} \tag{15}$$

We can solve two equations with two unknowns, which are τ_1 and τ_2. The answer is represented in Equation 16 where $\alpha = u \cdot u$, $\beta = u \cdot v, \zeta = v \cdot v, \xi = u \cdot (o_1 - o_2)$, and $\varepsilon = v \cdot (o_1 - o_2)$ respectively.

$$\begin{aligned} \tau_1 &= \frac{\beta\varepsilon - \zeta\xi}{\alpha\zeta - \beta^2} \\ \tau_2 &= \frac{\alpha\varepsilon - \zeta\xi}{\alpha\zeta - \beta^2} \end{aligned} \tag{16}$$

We use these points to retrieve depth to calculate accelerations observed by the camera using Equation 1, and compare it to the accelerations sensed by the wearable accelerometer.

5 Experimental Results

To our knowledge, this is the first work that attempts to match arbitrary movements observed between an accelerometer and cameras without using information about type of the movement or sensor placement. Hence, it is challenging to present a comparison to the state-of-the-art. In our experimental validation, we focus on determining the ability to find the matched sequences that stem from the same movements. We utilize Sony PS Eye cameras, with resolution of 640 ×480, and a frame rate of 30Hz. For the wearable accelerometers, we use a custom-designed wearable device developed by our research laboratory operating with a sampling rate of 60Hz [24]. It should be noted that any type of accelerometer embedded in commercial smartwatches or smartphones can work with our proposed algorithms.

5.1 Single-Camera Performance

We demonstrate the effectiveness of our proposed techniques in the context of the following experiment. One person is holding an accelerometer in his hand, and performs a movement or a gesture. Four other participants also perform gestures at the same time all captured by the same camera but they do not hold an accelerometer. This creates a paradigm where the camera sees multiple moving objects while just one of them is being measured by the accelerometer at the same time. The movement of four participants without an accelerometer does not necessarily need to be the same as the person with the accelerometer. We attempt to identify the movements associated with the person who is holding the accelerometer. This will demonstrate a simple application that can identify users in a space who provide their accelerometer streams, and when a match is determined, this would be used to provide certain services for the user.

Table 1. Transformed distance between cameras and accelerometers for six sets of experiments at constant depth

Dist.	No.1 Walking					No.2 Circling Clockwise					No.3 Circling Counterclockwise				
	A_1	A_2	A_3	A_4	A_5	A_1	A_2	A_3	A_4	A_5	A_1	A_2	A_3	A_4	A_5
C_1	**0.84**	35.01	39.13	19.60	13.19	**7.47**	46.80	82.39	72.25	88.98	**8.23**	43.79	105.66	95.45	88.63
C_2	32.88	**1.26**	9.54	19.44	22.42	41.64	**4.98**	82.46	66.70	72.62	40.65	**6.78**	105.88	95.33	88.49
C_3	33.90	13.99	**1.31**	19.06	22.43	73.48	77.06	**12.32**	72.50	90.64	58.16	61.60	**9.18**	95.73	88.61
C_4	34.03	45.21	39.91	**0.67**	22.33	70.83	70.02	79.45	**8.34**	89.85	58.12	61.04	105.85	**13.74**	88.59
C_5	12.91	45.08	40.10	19.66	**0.74**	72.29	62.04	82.57	75.22	**5.42**	58.54	61.62	105.93	95.82	**20.01**
	No.4 Sliding Vertically					No.5 Waving Hands					No.6 Moving Randomly				
C_1	**14.44**	144.80	212.16	200.10	148.18	**17.68**	176.55	90.93	117.77	122.96	**1.46**	28.01	25.50	41.14	30.75
C_2	144.22	**15.58**	213.02	185.35	148.72	130.80	**13.79**	120.57	117.84	127.94	11.89	**3.12**	25.55	36.76	30.77
C_3	145.29	153.76	**32.61**	200.63	131.20	113.37	176.44	**12.16**	117.97	127.63	12.01	28.21	**2.12**	42.95	15.30
C_4	146.33	145.31	213.01	**52.15**	148.38	129.68	176.40	120.45	**19.32**	127.35	11.33	23.52	25.77	**7.06**	30.72
C_5	139.22	153.75	160.88	200.13	**72.50**	130.30	176.66	120.25	117.35	**21.81**	11.99	28.21	12.85	42.72	**3.48**

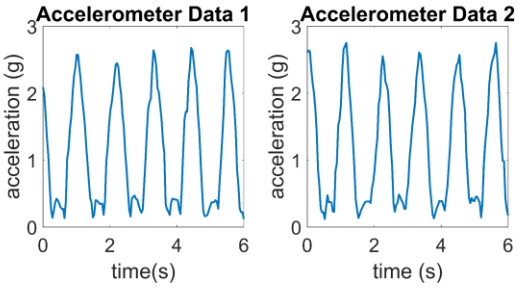

Figure 6. Two sequences of accelerometer data from experiment No.4 corresponding to two different subjects

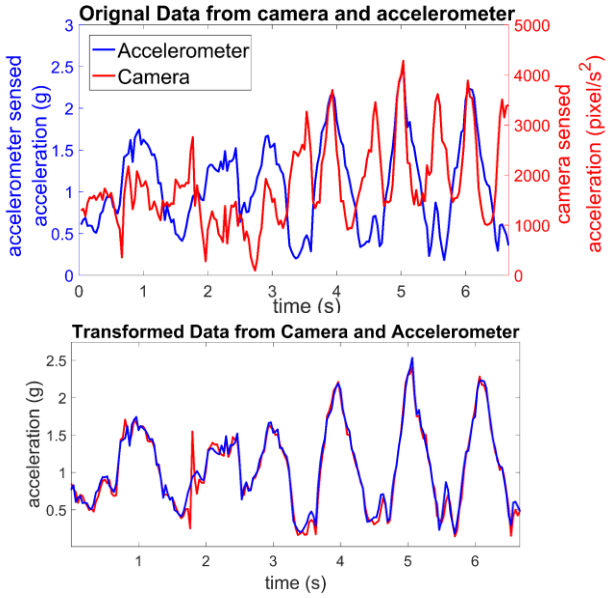

Figure 7. Data from accelerometers and cameras before and after CST was applied

To test the performance of our method, we conduct different sets of experiments while within each set, there are five participants performing a gesture/activity. Each set of experiments is repeated five times and in each repetition, one of the subjects holds an accelerometer while others do not hold any. Since it is challenging for a camera-based object tracking algorithm to track five moving participants at the same time with reasonable accuracy, we record each movement five times separately and manually bind them together into one experiment as in [4]. We capture each movement with 200 frames, which is 6.67 seconds since the cameras operate at 30 frames per second.

5.1.1 Constant Depth Scenario. We conduct six sets of experiments to evaluate the accuracy of the proposed system under constant depth motions. The six movements tested are walking, waving hands, circling the hand clockwise, circling the hand counterclockwise, sliding the hand vertically, and moving the hand randomly. The results of the six sets of experiments are shown in Table 1, where C_i in the table represents the moving object i tracked by the LK algorithm in the camera view, and A_i denotes the accelerometer that is being held by the subject i. Column i^{th} in the table shows a separate repetition of the experiment while i^{th} subject is holding the accelerometer and others do not. The distance between the acceleration calculated from the camera data for 5 tracked objects $(C_{1\text{-}5})$ and the acceleration measured by the accelerometer on subject i is shown in i^{th} column. Therefore, the element with the smallest value at each column corresponds to determining a match between the accelerometer and the corresponding object detected in the camera view. It is evident from Table 1 that the minimum transformed distances between these two modalities appear at the diagonal of the table, which indicates all correct matchings are determined by our proposed algorithm. Figure 6 offers a clear illustration on the similarity of the gestures performed by different participants in our experiment. Those two sequences of readings from accelerometer from two participants in the figure show a high similarity in both magnitude and pattern. However, our technique is still capable of matching the objects associated with an accelerometer in the camera frames correctly, and rejecting irrelevant sequence, after the proposed translation.

Figure 7 also shows how CST brings data from these two modalities into the same space. The figure demonstrates that prior to applying the CST method to the camera data, the two sequences of readings from the camera and the accelerometer do not offer similarities. Their patterns are different due to the impact of gravity on the accelerometer, and their magnitudes are dissimilar due to their representation with different physical units. However, the optimal CST successfully translates the readings from the camera space to the accelerometer physical space, where both modalities share the same pattern and physical units.

We derive additional observations from Table 1. First, when looking column-wise, the translated distance between unmatched pairs is similar in most cases, which validates our explanation in section 4.3.2 that the translated distance between unmatched pairs

is almost the same. Second, it is impossible to distinguish matched/unmatched pairs by utilizing a constant threshold. For example, the translated distance between unmatched C_i (i = 2, 3, 4, 5) and A_1 in experiment No. 6 is around 12, which is even smaller than the distance between A_5 and C_5 in experiment No. 4, which is 72.50, while they are matched. This indicates that the absolute value of translated distance of matched pairs is not necessarily always smaller than unmatched pairs. A more detailed comparison between our criteria and the constant threshold is offered in the next subsection.

5.1.2 Effect of Proposed Criteria. To further demonstrate that it is necessary to develop an alternative approach to the constant threshold selection and that our proposed criteria provides suitable separation for matched and unmatched sequences, we gather distance values in Table 1 and plot them in Figure 8. We calculate both translated distance and λ in Equation 9, and scattered results into two plots. The figure on the left indicates that by using only a constant threshold on the translated distance, it is impossible to separate matched and unmatched pairs apart, since it is clear that the red points (matched pairs) and black points (unmatched pairs) are not linearly separable. This further supports our hypothesis that the translated distance between two data sequences cannot distinguish matched and unmatched pairs apart since the translated distance is significantly affected by how fast the object is moving. The plot on the right indicates that our proposed method is able to transform all data points into a linear-separable space using Equation 9, and a visible difference between matched and unmatched pairs is observed.

5.1.3 Convergence Time. In our experiments the convergence time refers to the number of frames or samples needed for our algorithm to offer a correct decision in determining matched sequences. We calculate the number of frames needed to allow Equation 8 to converge. The experimental results regarding the convergence time for each object i are shown in Table 2, which indicates that on average 1.5s is sufficient for our algorithm to determine a match. That is, about 45 frames would be sufficient, since our cameras operate with a frame rate of 30 per second. O_i in the table corresponds to the convergence time in seconds for detecting the matched pair A_i and C_i in each experiment listed in Table 1. It should be noted that convergence time is largely affected by the quality of data; too much noise may cause longer convergence time and even result in a mismatch. The data in Table 2 is calculated based on our experimental setup and environment.

5.1.4 Single-Camera Depth Varying Performance. In one set of experiments, the participants were asked to perform hand gestures to test the performance of our technique under limited depth-varying conditions using a single camera. In this experiment, the participants were allowed to move their hand forward and backward but keeping the direction fixed. This leads to limited changing of the relative depth of the hand with respect to the camera. The analysis for the depth varying experiments are similar to the constant depth scenario since both scenarios share the similar concept and methodology.

Figure 9 demonstrates the impact of our proposed filter as shown in Equation 12 to eliminate the unobservable acceleration

Table 2. Convergence time in experiments

Convergence Time (s)	O_1	O_2	O_3	O_4	O_5
Experiment No.1	0.767	0.533	0.833	0.967	0.667
Experiment No.2	0.800	0.867	0.600	0.567	0.333
Experiment No.3	0.633	0.500	0.800	1.033	0.700
Experiment No.4	0.500	0.833	1.200	1.433	1.400
Experiment No.5	0.700	0.367	1.100	1.367	0.533
Experiment No.6	0.967	3.200	0.600	0.867	1.267

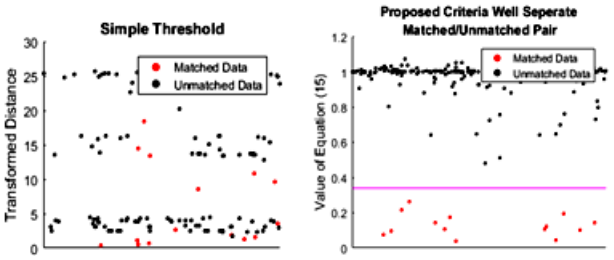

Figure 8. Comparison between naive criteria and proposed criteria to detect matched and unmatched sequences

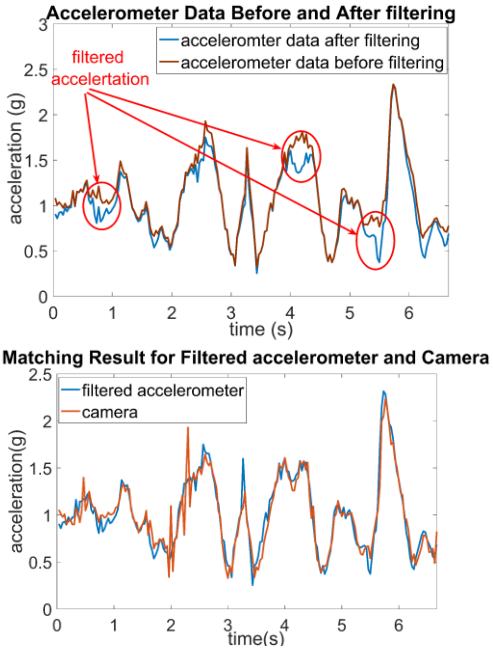

Figure 9. Match in depth varying scenario with one camera

from the camera, which assists our algorithms to establish a stronger relationship between the camera and the wearable accelerometer readings. In presence of the same experimental settings as in the constant depth scenario, Table 3 shows the translated distance calculated in our experiments with five wearable accelerometers moved by the participants. As demonstrated in Table 3, for all five participants, our algorithm was able to identify and match the participant with an accelerometer to the correct corresponding object detected by the camera under the assumption that the direction of the motions remains constant while the depth is changing. However, our method may fail in significant depth changes that include changes in the direction of the motions. Thus, in case of using a single camera for varying-depth scenarios the direction of the movement

Table 3. Translated distance when depth varies with one camera

Distance	A_1	A_2	A_3	A_4	A_5
C_1	3.00	6.08	16.52	10.80	36.51
C_2	9.40	2.17	17.54	10.39	38.03
C_3	8.55	6.16	9.10	10.41	29.86
C_4	9.28	6.02	17.33	3.11	37.88
C_5	9.22	6.22	13.22	10.46	6.94

Table 4. Translated distance using multiple cameras

Distance	Experiment No. 1				
	A_1	A_2	A_3	A_4	A_5
C_1	4.12	62.57	70.93	92.46	59.86
C_2	65.11	3.39	79.38	85.23	64.08
C_3	67.31	72.55	4.88	38.46	73.08
C_4	85.05	71.35	29.67	6.09	94.65
C_5	65.35	65.93	78.67	104.87	3.66
Distance	Experiment No. 2				
C_1	6.25	86.94	52.56	69.49	56.33
C_2	83.66	6.61	48.10	47.26	42.59
C_3	70.28	65.35	2.03	15.61	7.77
C_4	81.48	59.43	10.88	3.67	7.63
C_5	76.30	60.49	10.56	12.42	0.92

should be retained so that our method can add gravity interference to camera-sensed accelerations in a correct direction.

5.2 Multi-camera Performance

We use two cameras in our experiments to assess the performance of the proposed solution. We arbitrarily select one camera as the global frame. The relative angle of the other camera to the global frame is 60 degree around z-axis and the distance between the centers of the two cameras is 1.2 meters. Two sets of experiments are conducted to evaluate accuracy of multiple camera solution. Within each set, the subjects are asked to perform arbitrary hand gestures. Distance between different moving objects in each experiment is shown in Table 4 where all minimal distances appear at the diagonal of the table, which indicates that all the matched pairs were successfully determined by our method.

6 CONCLUSION

We proposed a method to tag the movement of an accelerometer in frames acquired by camera(s). The main contribution of our approach lies in the fact that the tagging does not require the knowledge of the orientation of the accelerometer or type of the movement. Our proposed approach retains the ability to match arbitrary movements in case of constant depth and with minimal constraints in case of varying depth using a single camera. We also offered a solution based on multiple cameras that works for any kind of motion without any limitations. We validated the performance of our proposed techniques and demonstrated their effectiveness in tagging an accelerometer in the view of a camera in no depth and minor depth changes. We also showed that this solution could be scaled to multiple cameras when more advanced tagging is required. Our proposed methodology will enable various applications that can benefit from tagging an accelerometer in the view of the camera, such as user identification or asset tracking.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation, under grants CNS-1734039 and ECCS-1509063. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations.

REFERENCE

[1] Rodgers, M. M., Pai, V. M., & Conroy, R. S., Recent advances in wearable sensors for health monitoring. IEEE Sensors Journal 2015, 15(6), 3119-3126.
[2] Akbari, A., Thomas, X., & Jafari, R., Automatic noise estimation and context-enhanced data fusion of IMU and Kinect for human motion measurement, In BSN, 2017, IEEE 14th International Conference on (pp. 178-182). IEEE.
[3] Guenterberg, E., Ghasemzadeh, H., Loseu, V., & Jafari, R. (2009, June). Distributed continuous action recognition using a hidden markov model in body sensor networks. In International Conference on Distributed Computing in Sensor Systems (pp. 145-158). Springer, Berlin, Heidelberg.
[4] Jung, D., T. Teixeira, and A. Savvides. Towards cooperative localization of wearable sensors using accelerometers and cameras. in INFOCOM, IEEE, 2010
[5] Barron, J.L., D.J. Fleet, and S.S. Beauchemin, Performance of optical flow techniques. International journal of computer vision, 1994. 12(1): p. 43-77.
[6] Bouguet, J.-Y., Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. Intel Corporation, 2001. 5(1-10): p. 4.
[7] Chen, C., Jafari, R., & Kehtarnavaz, N. (2016, March). Fusion of depth, skeleton, and inertial data for human action recognition. In Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on (pp. 2712-2716). IEEE.
[8] Wu, J., & Jafari, R. (2017). Seamless Vision-assisted Placement Calibration for Wearable Inertial Sensors. ACM Transactions on Embedded Computing Systems (TECS), 16(3), 71.
[9] Von Marcard, T., Pons-Moll, G., & Rosenhahn, B., Human pose estimation from video and imus. IEEE transactions on pattern analysis and machine intelligence, 2016,
[10] Chen, C., Jafari, R., & Kehtarnavaz, N. (2016). A real-time human action recognition system using depth and inertial sensor fusion. IEEE Sensors Journal, 16(3), 773-781.
[11] Mur-Artal, R., & Tardós, J. D., Visual-inertial monocular SLAM with map reuse, IEEE Robotics and Automation Letters, 2017, 2(2), 796-803.
[12] Mur-Artal, R., & Tardós, J. D.. Orb-slam2: An open-source SLAM system for monocular, stereo, and rgb-d cameras. IEEE Transactions on Robotics, 2017, 33(5).
[13] Kelly, J. and G.S. Sukhatme, Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration. The International Journal of Robotics Research, 2011. 30(1): p. 56-79.
[14] Strelow, D. and S. Singh, Motion estimation from image and inertial measurements. International Journal of Robotics Research, 2004. 23(12): p. 1157-1195.
[15] Martinelli, A., Vision and IMU data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination. IEEE Transactions on Robotics, 2012. 28(1): p. 44-60.
[16] Chen, K., Li, T., Kim, H. S., Culler, D. E., & Katz, R. H. (2018, November). MARVEL: Enabling Mobile Augmented Reality with Low Energy and Low Latency. In Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems (pp. 292-304). ACM.
[17] Hol, J.D., et al., Sensor fusion for augmented reality, 9th International Conference on Information Fusion. 2006. IEEE.
[18] Shigeta, O., S. Kagami, and K. Hashimoto. Identifying a moving object with an accelerometer in a camera view. in Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on. 2008. IEEE.
[19] Maki, Y., Kagami, S., & Hashimoto, K., Accelerometer detection in a camera view based on feature point tracking. In System Integration (SII), 2010 IEEE/SICE International Symposium on (pp. 448-453). IEEE.
[20] Hartley, R., & Zisserman, A., Multiple view geometry in computer vision., 2003, Cambridge university press.
[21] Björck, Å., Numerical methods for least squares problems. 1996: SIAM.
[22] Chan, A., et al., Comparison of two-dimensional vs three-dimensional camera systems in laparoscopic surgery. Surgical endoscopy, 1997. 11(5): p. 438-440.
[23] Jang, W., et al., Structured-light stereo: Comparative analysis and integration of structured-light and active stereo for measuring dynamic shape. Optics and Lasers in Engineering, 2013. 51(11): p. 1255-1264.
[24] Bennett, T. R., Massey, H. C., Wu, J., Hasnain, S. A., & Jafari, R. (2016). MotionSynthesis Toolset (MoST): An Open Source Tool and Data Set for Human Motion Data Synthesis and Validation. IEEE Sensors Journal, 16(13), 5365-5375.

7 Appendices

7.1 Matching Criteria Formulation

Our proposed decision criteria is based on the fact that function h (Equation 8) will have a global minimum value when a_i^c and a_i^m are matched, while it tends to decrease smoothly when a_i^c and a_i^m are not matched or are not coming from the same motion. Treating $\|g_{cam}\|$ as a whole to simplify our analysis, the function h can be written as Equation 17.

$$h(\|g_{cam}\|) = \sum_{j=1}^{n} (\| \tilde{a}_{i,j}^c \| \cdot \frac{\|g_{phy}\|}{\|g_{cam}\|} - \|a_{i,j}^m\|)^2 = \sum_{j=1}^{n} (\gamma_j(\|g_{cam}\|) - \|a_{i,j}^m\|)^2 \quad (17)$$

The function $\gamma_j(\|g_{cam}\|)$ can be considered as the translation function to translate the camera's data $\| \tilde{a}_{i,j}^c \|$ to m/s^2 by applying different values of CST calculated from corresponding $\|g_{cam}\|$. The derivative of function h in Equation 17 with respect to $\|g_{cam}\|$ is as shown in Equation 18:

$$h'(\|g_{cam}\|) = 2\sum_{j=1}^{n} (\gamma_j(\|g_{cam}\|) - \|a_{i,j}^m\|) \cdot \gamma_j'(\|g_{cam}\|) \quad (18)$$

A heuristic explanation for the property of h' is as follows: for two matched sequences, there should exist an optimal CST, ω, to translate the two modalities so that the two sequences become most similar. This optimal value of ω leads to the value of $\sum_{j=1}^{n}(\gamma_j(\|g_{cam}\|) - \|a_{i,j}^m\|)$ in Equation 18 approaching zero, turning $h'(\|g_{cam}\|)$ to zero, and function h reaching its global minimum value. For two unmatched sequences, however, it is unlikely to obtain such a CST, and thus $h'(\|g_{cam}\|)$ approaches zero as $\|g_{cam}\| \to \infty$ since $lim_{\|g_{cam}\|\to\infty} \gamma_j'(\|g_{cam}\|) = 0$. Therefore, the typical shape of function h in Equation 17 for both matched and unmatched pairs of sequences would be same as Figure 10, which is consistent with our experimental validation.

Using this property, we formulate our criteria as follows:

$$\begin{cases} \lambda < \rho & \rightarrow \ matched\ sequences \\ \lambda \geq \rho & \rightarrow \ irrelevant\ sequences \end{cases}$$

Where

$$\lambda = \frac{min(h(\|g_{cam}\|))}{\mu}$$

$$\mu = \lim_{\|g_{cam}\|\to\infty} f(\|g_{cam}\|) = \sum_{t=t_1}^{t_2} \left(9.8 - \|a_{i,j}^m\|\right)^2$$

where $\rho \leq 1$ is a constant value specified by the user and serves as a threshold, but λ is calculated based on the camera and accelerometer data. The proposed criteria adjusts itself according to the acceleration of movements because it takes advantage of readings from the accelerometer to adjust the value of μ. Fast movements lead to a larger value of μ. The reason is that $a_{i,j}^m$ consists of both gravity and linear acceleration, so if the object is not moving, $a_{i,j}^m$ will be equal to gravity and the term $9.8 - a_{i,j}^m$ will be zero. However, if the object experiences high acceleration, $a_{i,j}^m$ will be different from 9.8 and μ will be larger, thus the value of $min\, h(\|g_{cam}\|)$ will be allowed to be larger to satisfy Equation 11.

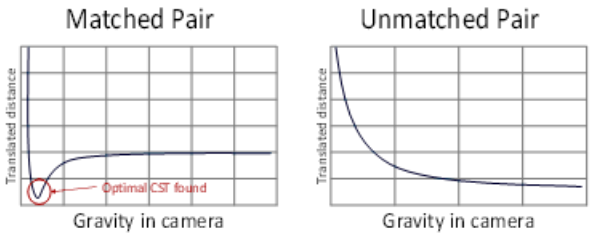

Figure 10. Different shapes of function h when the pair of data is matched or unmatched

7.2 Rotation Matrix

The value of rotation matrix $\boldsymbol{R}$ relative to the global frame in Equation 13 and Equation 14 is calculate by Equation 19.

$$\boldsymbol{R} = \boldsymbol{R_z}(\psi)\boldsymbol{R_y}(\theta)\boldsymbol{R_x}(\varphi)$$

where

$$\boldsymbol{R_z}(\psi) = \begin{bmatrix} cos\psi & -sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$\boldsymbol{R_y}(\theta) = \begin{bmatrix} cos\theta & 0 & sin\theta \\ 0 & 1 & 0 \\ -sin\theta & 0 & cos\theta \end{bmatrix} \quad (19)$$
$$\boldsymbol{R_x}(\varphi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\varphi & -sin\varphi \\ 0 & sin\varphi & cos\varphi \end{bmatrix}$$

where ψ (roll), θ (pitch), and φ (roll) are Euler angles.

7.3 System Demonstration

The following link contains a video that demonstrates the operation of our proposed algorithm: http://tiny.cc/n7dvky. There are four participants in the video waving their hands. Figure 11 shows a screen shots of this video. The participant on the right is holding an accelerometer. The accelerometer is tracked successfully and is marked with green dots while all other movements are marked by red dots.

Figure 11. Screen shots of the video showing how the proposed algorithm tags the person carrying an accelerometer in the camera scene